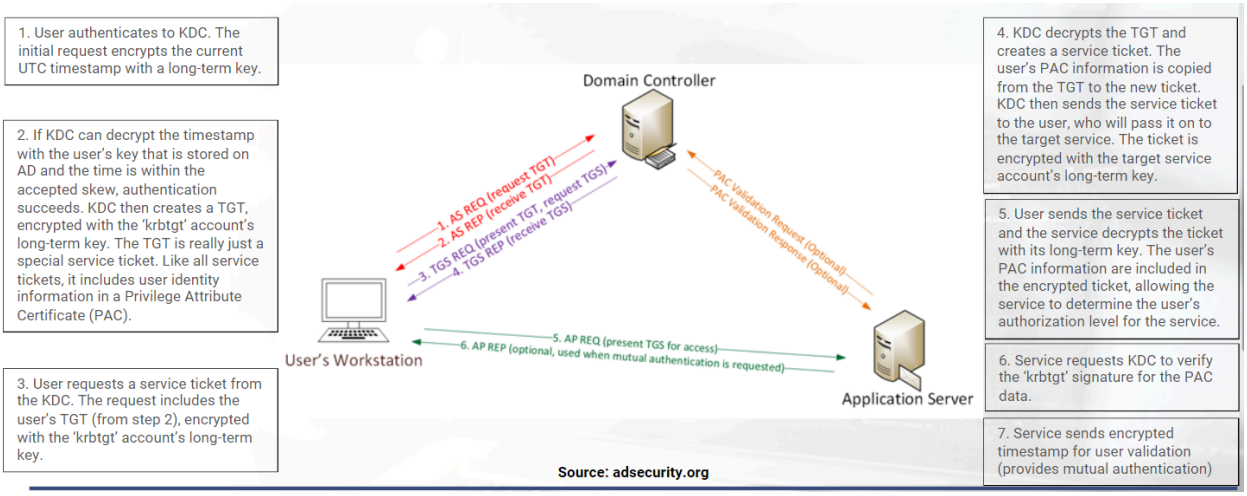
Pass the ticket attack:

Attackers TTPs are evolving, since most people moved on from NTLM to Kerberos, the attacker came up with Pass the ticket attack. The Kerberos is Ticketing protocol, mean when someone authenticate , they get a ticket that they can use to access services and that's what makes it SSO (single sign on) as the user logs in once and gets a ticket which then use that ticket to access services, so it dosnt login each time it to access something in the domain like the NTLM, we just use that ticket we get when we login.

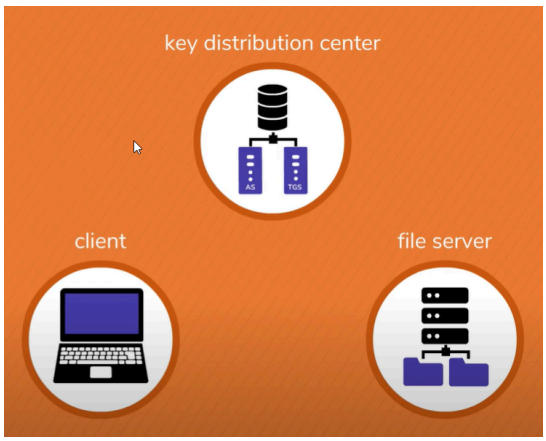
Like the Pass the hash, pass the ticket is also used for lateral movement.

Lets first see how Kerberos authentication works:



In Kerberos environment, there is 3 component:

1. Client
2. Key distribution center that has the authentication server (AS) and the Ticket Grant System (TGS) in it
3. Resource like Files server that user wants to access.



This is how the authentication process works:

1. **Authentication Request to the AS (Not Encrypted)**:
    - The user sends their username in plain text to the AS (authentication server). This step is not encrypted because the AS needs to identify the user. Yea it just sends the username, no password or its hash is sent.
2. **Response from the AS (Encrypted)**:
    - After receiving the username, it checks its database to see if the user accounts exist, then the AS (authentication) generates the Ticket Granting Ticket (TGT) for the user.
    - The TGT is encrypted with a secret key derived from the user's password (or password hash), making it secure.
3. **TGT Response (Encrypted)**:
    - The AS sends the TGT back to the user, and this transmission is encrypted.
    - The TGT is encrypted with the user's password-derived key, ensuring that only the user can decrypt it.
4. TGT storage:
    - Then the Ticket received by the user and stored locally in Memory, not on the disk.
5. TGT request to TGS (ticket granting service):
    - Now the user has the ticket and when it wants to access a services or a resources, it will send its ticket to the TGS, so the TGT grant the user the access they need. In this request to the TGS, the user puts the Service name they want to access, and then the TGS check if the user should access the service, like check the user privileges and then grant them the access. Mean the TGS validates the TGT ticket, like check the ticket authenticity and permissions. And then the TGS will generate a TGS ticket for the target service the user wants to access. So the user logs in and it gets the TGT and then it uses that TGT to access services by using it to request TGS for the access, and if the access is permitted the user will get specific to that service another ticket called TGS ticket.
    - In this ticket also contains sessions keys which is the service secret key that will be used for the encryption between the service and the client.
6. Then the client use that TGS to access that service:
    - Once the client receive the TGS ticket, it will send to the target service to access it, then the service will decrypt the ticket, and validate it.
7. Session establishment between the client and service:
    - Then the client and the service will establish a secure connection by sharing keys with each other. And this connection is tamper free, encrypted.

The Kerberos it self is secure as we see, all the connections are secure and everything is validate during the process. But there is one weakness and that is the ticket, mean if an attacker somehow manages to get that ticket, it can resend it to gain access to things like service, files…, both the TGT ticket and the TGS ticket, mean there is no identifying information in the ticket regarding the computer the ticket came from, it just validate the ticket is legit and that's it.

One of the ways that we can get the TGT ticket is to grab it from the memory, as we said the tickets are not stored on the drive but in memory so if we gain access to the system we can extract the ticket from its memory.

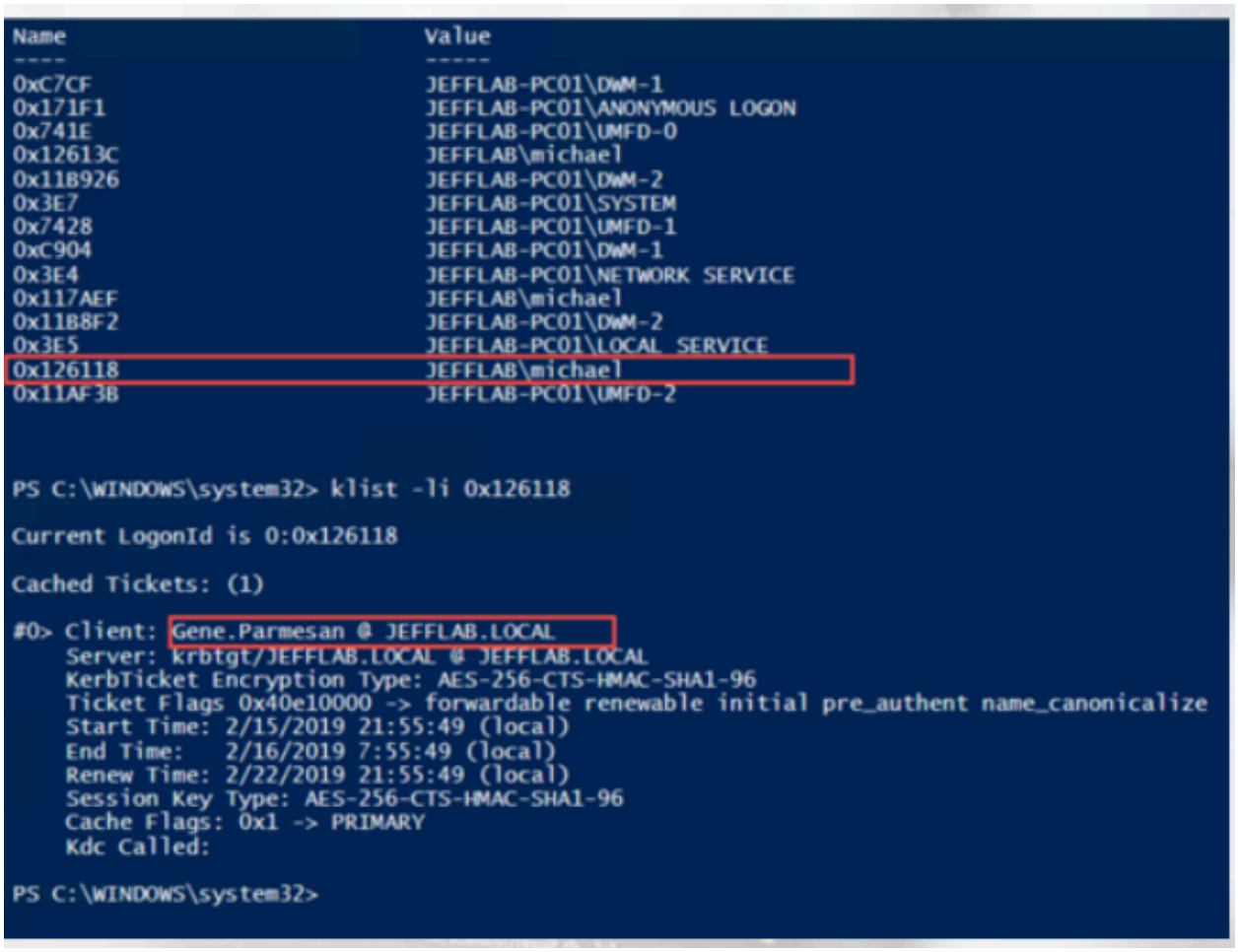
The authentication is handled by LSASS process of the windows, so its in the memory location of the LSASS process and one tool that can do this is Mimikatz.

Detection:

The attackers will more likely harvest the Tickets and then use them from another machine.

To detect pass the ticket attack on an endpoint is:

1. First list all the logon sessions on an endpoint, like we know we first login to the local machine, so we check all the users that logged in this machine, and then we get each login's Login ID that is in hex format. Like if this machine belongs to John we would see its log on session and from it we can get the Logon ID.
2. Then for each of the logon ID we will list its Kerberos ticket, like john was logged in and we got its Logon ID and we will check what tickets it used in that session or what tickets are associated with the Johns session.
3. Then we will identify tickets that don't match the user associated with session. Like we see the Session belong to the John and the ticket that is used belong to Kim. And that would be the Pass the ticket attack.



Here we first listed all the sessions, and we see many logon sessions like some belong to the services as they also logs in and have their own session, we take each one of those sessions logon ID. And then we used the klist -li LogOnID command to check the associated or used tickets in that user session, to get the cached tickets of that session, here we see the sessions belong to the Michael but the Ticket that is cached belong a user called Gene.Parmesan. mean the user Michael used a ticket that belong to Gene.Parmesan user.

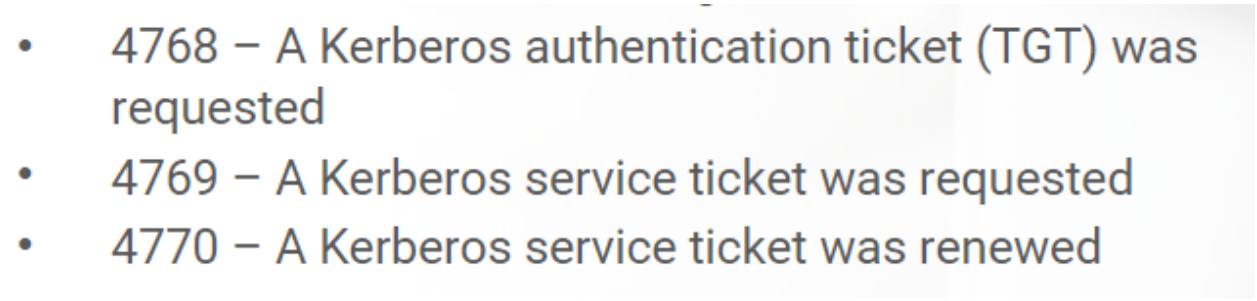
Detection on Domain controller:

In the Domain controller The first event we should see is a 4768 event. This is the TGT request and is the first thing that must happen for a user to leverage Kerberos to access a network resource. we will get one of these for each user for every endpoint they access our domain from. If a user account logs in from two separate workstations, they will request a TGT from each.

as we said The attackers will more likely harvest the Tickets and then use them from another machine, so the attacker will not request any TGT as they steal them and use it to request TGS or to renew that TGT using the TGT. so if we see TGS request or the TGT is renewed, we can look back at this TGT when it was

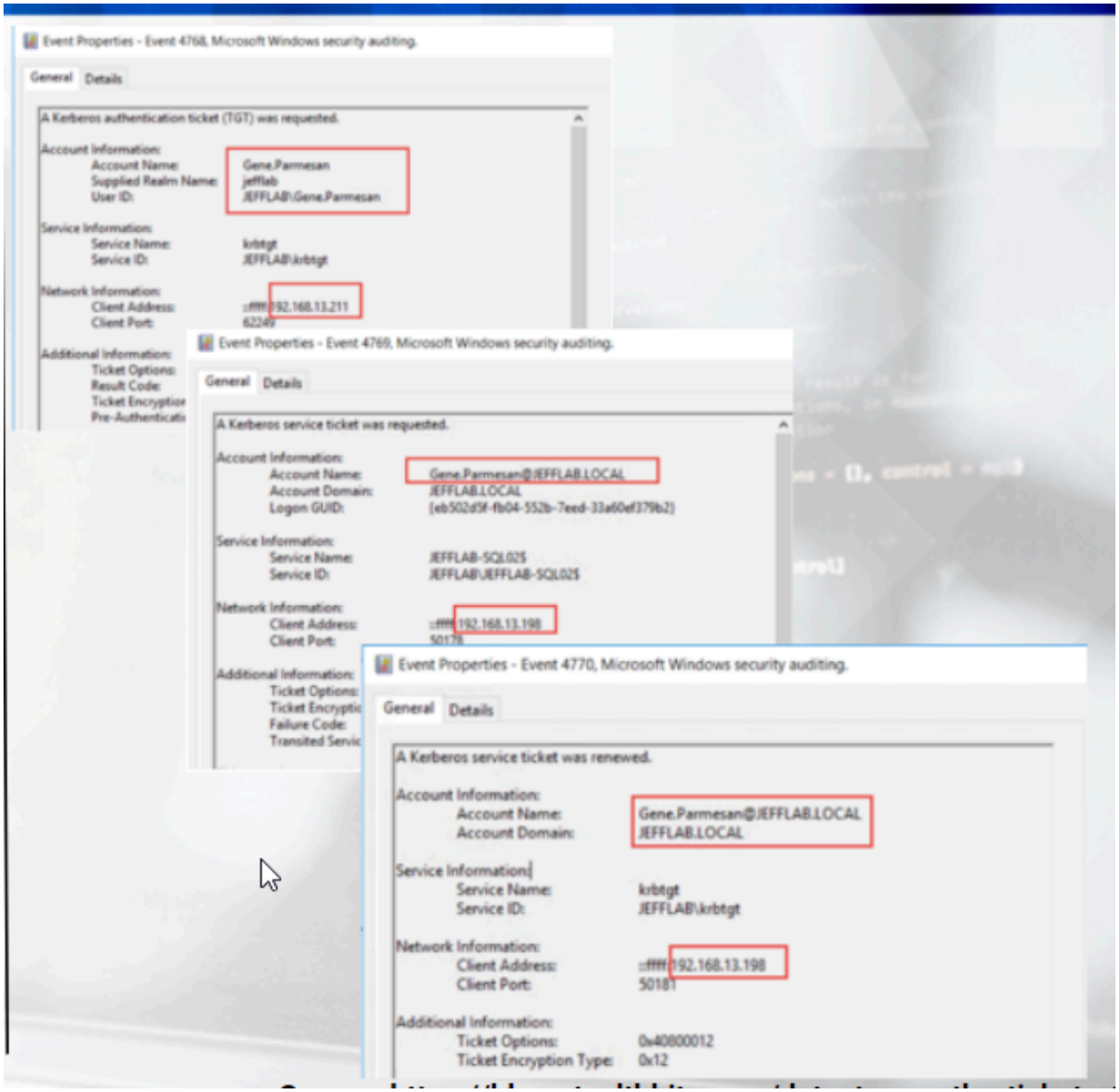
requested and see the Client IP and the User name that requested this TGT and then compare it to see if it matches or not.

these are the events that we are interested in:



So if once a TGT is stolen, and resent, we will see that the TGT is coming from a deferent Client IP where it was generated for. So like if IP *.*.*.92 authenticated and got a TGT ticket and then we see an IP *.*.*.134 used that ticket to request TGS to get access to a service, mean the ticket was given to the 92 client and then it was used by the 134 to get access to service. So for this we can use the event ID 4769 to see all the requested TGS, and then we check the TGS to see from which IP it was requested, and then we use the ID 4768, to check which IPs authenticated and got a TGT, and then compare it to find if any IP that got a TGT and then it was used by a deferent IP to get the TGS ticket.

But attackers are also smart and they will renew the ticket, but this can also be detect like if we see with event ID 4768 which users are logged and got a ticket, and then we check which IPs renewed the tickets with event ID 4770, so if we see a user authenticate like with IP *.*.*.92 and then its ticket was renewed from IP *.*.*.134, then it would be suspicious.



Here first we see the account JEFFLAB.LOCAL authenticated from IP 211 and got a ticket as we see the event is 4768 which means a TGT ticket was requested, and then we see the same ticket that was given to the IP 211 is used from deferent IP 198, as we see in the event ID 4769 which means TGS is requested, mean a service was requested using that TGT that belong to JEFFLAB.LOCAL user but from a deferent IP that it was given to during the authentication, and that is suspicious and then we see the same IP 198 which is a suspected IP renewed the ticket as we see the event 4677 which means the Kerberos ticket was renewed.